How Does the Smoothness Approximation Method Facilitate Generalization for Federated Adversarial Learning?

Wenjun Ding[1,2], Ying An[3], Lixing Chen[4,5], Shichao Kan[1], Fan Wu[1,*], and Zhe Qu[1,2,*]

[1]School of Computer Science and Engineering, Central South University, Changsha, China
[2]Xiangjiang Laboratory, Changsha, China
[3]Big Data Institute, Central South University, Changsha, China
[4]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China
[5]Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai, China
{234712240, anying, kanshichao, wfwufan, zhe_qu}@csu.edu.cn, lxchen@sjtu.edu.cn

Abstract

Federated Adversarial Learning (FAL) is a robust framework for resisting adversarial attacks on federated learning. Although some FAL studies have developed efficient algorithms, they primarily focus on convergence performance and overlook generalization. Generalization is crucial for evaluating algorithm performance on unseen data. However, generalization analysis is more challenging due to non-smooth adversarial loss functions. A common approach to addressing this issue is to leverage smoothness approximation. In this paper, we develop algorithm stability measures to evaluate the generalization performance of two popular FAL algorithms: Vanilla FAL (VFAL) and Slack FAL (SFAL), using three different smooth approximation methods: 1) Surrogate Smoothness Approximation (SSA), (2) Randomized Smoothness Approximation (RSA), and (3) Over-Parameterized Smoothness Approximation (OPSA). Based on our in-depth analysis, we answer how to properly set the smoothness approximation method to mitigate generalization error in FAL. Moreover, we identify RSA as the most effective generalization error reduction method. In highly data-heterogeneous scenarios, we also recommend employing SFAL to mitigate the deterioration of generalization performance caused by heterogeneity. Based on our theoretical results, we provide insights to help develop more efficient FAL algorithms, such as designing new metrics and dynamic aggregation rules to mitigate heterogeneity.

Introduction

Federated Learning (FL) (McMahan et al. 2017; Qu et al. 2022; Li et al. 2023) plays an important role as it allows different clients to train models collaboratively without sharing data samples. Although FL is considered a secure paradigm to protect users' private data, it has been vulnerable to adversarial attacks (Wang et al. 2020; Bagdasaryan et al. 2020). Adversarial examples (Szegedy et al. 2013; Goodfellow, Shlens, and Szegedy 2014) are typically designed to mislead models into producing incorrect outputs, significantly degrading learning performance and achieving the attacker's goals.

To counter these attacks, much effort has been made to improve neural networks' resistance to such perturbations using Adversarial Learning (AL) (Goodfellow, Shlens, and Szegedy 2014; Samangouei, Kabkab, and Chellappa 2018; Madry et al. 2018). Generating adversarial examples during neural network training is one of the most effective approaches for AL (Carlini and Wagner 2017; Athalye, Carlini, and Wagner 2018; Croce and Hein 2020). Consequently, recent studies (Zizzo et al. 2020; Hong et al. 2021; Shah et al. 2021; Zhou et al. 2022; Li et al. 2021) have proposed a new FL framework called Federated Adversarial Learning (FAL) to mitigate the impact of adversarial attacks. It is important to improve the robustness of FL in real-world applications.

There are two popular algorithms for FAL: *Vanilla FAL (VFAL)* (Shah et al. 2021) and *Slack FAL (SFAL)* (Zhu et al. 2023). In particular, VFAL combines FedAvg (McMahan et al. 2017) with AL on the client side to train the global model. In contrast, SFAL modifies the aggregation process by dynamically adjusting the weights of certain client models during aggregation based on their evaluated importance.

Although these two algorithms have shown significant effectiveness and robustness, they prioritize convergence at the expense of generalization ability (Li, Song, and Yang 2023; Zhang et al. 2022). Generalization is a crucial aspect of evaluating an algorithm, as it measures the performance of trained models on unseen data. One popular approach to studying generalization is to examine algorithmic stability (Bousquet and Elisseeff 2002; Hardt, Recht, and Singer 2016), which measures sensitivity to perturbations in the training dataset. Recently, a series of studies (Xing, Song, and Cheng 2021a; Xiao et al. 2022a; Cheng, Fu, and Farnia 2024) have investigated the generalization error in AL.

Unfortunately, these generalization studies primarily focus on the centralized setting. In FAL, one of the most important problems is the heterogeneity across clients, which leads to locally generated adversarial examples being highly biased to each local distribution (Zhang et al. 2023). Experimental results have demonstrated that this exacerbated data impairs the generalization ability of FAL. Thus, the generalization bounds of AL are insufficient for generalizing to FAL. While (Sun, Niu, and Wei 2023; Lei, Sun, and Liu 2023) explored the relationship between data heterogeneity and generalization in FL, characterizing generalization in FAL poses additional challenges.

*Corresponding Authors.
Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Unlike FL, the loss function in FAL (Sadeghi et al. 2020; Zhou et al. 2022) is typically non-smooth, violating the stability analysis of algorithms that rely on smooth functions (Hardt, Recht, and Singer 2016; Sun, Niu, and Wei 2023; Lei, Sun, and Liu 2023). A practical way to tackle this issue is to employ a method of smoothness approximation, e.g., (1) *Surrogate Smoothness Approximation (SSA)* (Cui et al. 2021), (2) *Randomized Smoothness Approximation (RSA)* (Alashqar et al. 2023), and (3) *Over-Parameterized Smoothness Approximation (OPSA)* (Li, Song, and Yang 2023; Li et al. 2022). Although these methods can mitigate the non-smoothness property, generating uncertain adversarial examples may still lead the global model into sharp valleys and reduce the consistency of FAL, hindering its generalization performance. This raises the question: **How to properly set the smoothness approximation method to reduce the generalization error for FAL?**

The answer to the above question is two-fold: 1. Which approximation method is most suitable for FAL? 2. How should the parameters of each approximation method be set to reduce the generalization error? Therefore, in this paper, we provide detailed analyses for VFAL (Zizzo et al. 2020; Shah et al. 2021; Li et al. 2022; Zhang et al. 2022) and SFAL (Zhu et al. 2023) using the three smoothness approximation methods. To the best of our knowledge, this is the first study to investigate the generalization of FAL. Our findings will offer valuable insights for developing efficient FAL algorithms. The main contributions can be summarized as follows:

(1) We demonstrate the generalization bound of the three smoothness approximation methods for the VFAL algorithm. Our results first indicate that increased heterogeneity significantly impairs the ability to generalize. Then, the result of RSA shows the best generalization error, scaling with $\mathcal{O}(T)$. Notably, SSA shows improved performance by adding less noise. For OPSA, properly controlling the width of the neural network can reduce the generalization error. (2) We also provide the generalization bound of SFAL and compare it with VFAL. Our analysis demonstrates the impact of different global aggregation methods on generalization bound. In particular, considering some AL-related metrics, i.e., local adversarial loss, as a client contribution evaluation criterion to design dynamic global aggregation rules helps to improve generalization. (3) Based on our results, we have proposed some new metrics related to adversarial loss, e.g., contrastive loss and adversarial penalty, that may not have been considered in previous work. We believe that incorporating these metrics for dynamic global aggregation or into the local training could help design a more efficient FAL algorithm.

Related Work

FL is considered an efficient and privacy-preserving method for distributed learning environments. Generally, FL can be categorized into aggregation schemes (McMahan et al. 2017; Qu et al. 2022) and optimization schemes (Mohri, Sivek, and Suresh 2019; Reddi et al. 2020). However, FL remains vulnerable to adversarial attacks. Adversarial examples pose a significant threat to learning models, as perturbations in input data can mislead classifiers. To combat this, AL has been proposed to bolster robustness (Goodfellow, Shlens, and Szegedy 2014; Samangouei, Kabkab, and Chellappa 2018; Madry et al. 2018). Recent studies have also delved into robust overfitting (Rice, Wong, and Kolter 2020). However, directly integrating AL into FL poses several challenges, including poor convergence, data heterogeneity, and communication costs. To address these challenges, (Zizzo et al. 2020; Shah et al. 2021; Zhang et al. 2022; Li, Song, and Yang 2023; Zhu et al. 2023) have proposed corresponding FAL algorithms.

Generalization analysis has been widely used to evaluate algorithms in both FL and AL. (Mohri, Sivek, and Suresh 2019) leverages Rademacher complexity to develop a uniform convergence bound for FL. Subsequently, various FL studies have introduced generalization bounds based on Rademacher complexity (Sun and Wei 2022; Qu et al. 2023). (Hu, Li, and Liu 2022) considers a faster rate with Bernstein conditions and bounded losses under convex objectives. Additionally, (Sun, Niu, and Wei 2023; Lei, Sun, and Liu 2023) explore generalization upper bounds using on-average stability (Kearns and Ron 1997; Kuzborskij and Lampert 2018). Moreover, (Xing, Song, and Cheng 2021a) explores stability by highlighting the non-smooth nature of adversarial loss, and (Xiao et al. 2022a) develops stability bounds using smoothness approximation. Building on these insights, (Xiao et al. 2022b; Xing, Song, and Cheng 2021b) propose smoothed versions of SGDmax and robust deep neural networks.

Preliminaries

Federated Adversarial Learning (FAL)

To improve the robustness of FL against adversarial attacks, some studies have developed the FAL framework (Zizzo et al. 2020; Shah et al. 2021; Zhang et al. 2022; Li, Song, and Yang 2023; Zhu et al. 2023). Although these frameworks have demonstrated efficiency from the convergence perspective, their generalization performance may limit their applicability, which motivates us to analyze the underlying issues.

Typically, we consider a FAL framework with m clients. Each client i obtain the local dataset $(x_i, y_i) \in \mathcal{S}_i$ coming from the unknown distribution $P_i (P_i \neq P_{i'}, i \neq i')$ with the size $n_i = |\mathcal{S}_i|$. Let $\ell(\theta, (x, y))$ be the loss function with the learning model θ. The local objective of each client i is to minimize the local population risk, which is defined by:

$$R_i(\theta) = \mathbb{E}_{(x_i,y_i)\sim P_i}[\ell\left(f_\theta[x_i + A_\rho(f_\theta, x_i, y_i)], y_i\right)], \quad (1)$$

where A_ρ is an attack of strength $\rho > 0$ and intended to deteriorate the loss in the following way:

$$A_\rho(f_\theta, x_i, y_i) := \underset{\delta\in B_p(0,\rho)}{\operatorname{argmax}} \{\ell(f_\theta(x_i + \delta), y_i)\}, \quad (2)$$

where $B_p(0, \rho)$ is a $\mathcal{L}_p$ ball with radius ρ and $p = 1, 2$ or, ∞ for different types of attacks. For simplicity, we rewrite $\ell\left(f_\theta[x + A_\rho(f_\theta, x, y)], y\right) = \ell_\rho(\theta, z)$ and $z = \{x, y\}$ in this paper. Beyond the individual local objectives, all m clients collaboratively minimize the global objective, defined as $R(\theta) = \frac{1}{m}\sum_{i=1}^{m} R_i(\theta)$. However, directly minimizing the global objective $R(\theta)$ is challenging due to the unknown distributions P_i. Thus, a common way to approach $R(\theta)$ is to minimize the following global empirical risk:

$$R_{\mathcal{S}}(\theta) := \frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{S}_i}(\theta) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n_i}\sum_{j=1}^{n_i}\ell_\rho\left(\theta; z_{i,j}\right), \quad (3)$$

where $R_{\mathcal{S}_i}(\theta)$ is the local empirical risk $R_{\mathcal{S}_i}(\theta) := \frac{1}{n_i}\sum_{j=1}^{n_i}\ell_\rho(\theta; z_{i,j})$ with the local sample dataset $\mathcal{S}_i$. To investigate the generalization error in FAL, we focus on analyzing its algorithmic stability, which evaluates how sensitive a model is to changes in its training data. In the context of FAL, where the training dataset is distributed across various clients, it is crucial to assess how perturbations in the data at each client level affect the global model's stability.

Stability and Generalization

The generalization adversarial risk $\varepsilon_{gen}(\theta)$ is defined as the difference between population and empirical risk, i.e., $\varepsilon_{gen} := R(\theta) - R_{\mathcal{S}}(\theta)$. For a potentially randomized algorithm $\mathcal{A}$ that takes a dataset $\mathcal{S}$ as input and outputs a random vector $\theta = \mathcal{A}(\mathcal{S})$, we can define its expected generalization adversarial risk over the randomness of a training set $\mathcal{S}$ and stochastic algorithm $\mathcal{A}$ as follows:

$$\varepsilon_{gen}(\mathcal{A}) = \mathbb{E}_{\mathcal{S},\mathcal{A}}[R(\mathcal{A}(\mathcal{S})) - R_{\mathcal{S}}(\mathcal{A}(\mathcal{S})]. \tag{4}$$

One of the most popular ways to approach (4) is to consider the on-average stability (Kearns and Ron 1997; Shalev-Shwartz et al. 2010; Kuzborskij and Lampert 2018). Moreover, in our focused FAL, the generalization adversarial risk can be decomposed as $\varepsilon_{gen}(\mathcal{A}) = \mathbb{E}_{\mathcal{S},\mathcal{A}}[\frac{1}{m}\sum_{i=1}^{m}(R_i(\mathcal{A}(\mathcal{S})) - R_{\mathcal{S}_i}(\mathcal{A}(\mathcal{S})))]$, which implies that the generalization ability of the server is related to each client. Therefore, we are interested in the change in algorithm performance when one data sample is perturbed in any client. We introduce the following definitions related to on-average stability for the FAL framework, which is similar to those in (Sun, Niu, and Wei 2023; Lei, Sun, and Liu 2023).

Definition 1 (Neighboring Datasets)**.** *Given the entire dataset $\mathcal{S} = \cup_{i=1}^{m}\mathcal{S}_i$, where $\mathcal{S}_i$ is the local dataset of the i-th client with $\mathcal{S}_i = \{z_{i,1}, \ldots, z_{i,n_i}\}$, $\forall i \in [m]$, another dataset is called as neighboring to $\mathcal{S}$ for client i', denoted by $\mathcal{S}^{(i')}$, if $\mathcal{S}^{(i')} := \cup_{i\neq i'}\mathcal{S}_i \cup \mathcal{S}'_{i'}$, where $\mathcal{S}'_{i'} = \{z_{i',1}, \ldots, z_{i',j-1}, z'_{i',j}, z_{i',j+1}, \ldots, z_{i',n_i}\}$ with $z'_{i',j} \sim P_{i'}$, $\forall j \in [n_{i'}]$. And we call $z'_{i',j}$ the perturbed sample in $\mathcal{S}^{(i')}$.*

Definition 2 (On-Average Stability for FAL)**.** *A randomized algorithm $\mathcal{A}$ is ϵ-on-average stability if given any two neighboring datasets $\mathcal{S}$ and $\mathcal{S}^{(i')}$, then*

$$\max_{j\in[n_i]} \mathbb{E}_{\mathcal{A},\mathcal{S},z'_{i',j}}|\ell_\rho(\mathcal{A}(\mathcal{S}); z'_{i',j}) - \ell_\rho(\mathcal{A}(\mathcal{S}^{(i')}); z'_{i',j})| \leq \epsilon,$$

where $z'_{i',j}$ is the perturbed sample in $\mathcal{S}^{(i')}$, $\forall i \in [m]$.

On-average stability basically means any perturbation of samples across all clients cannot lead to a big change in the model trained by the algorithm in expectation. Moreover, we need to state a global assumption and a key lemma to indicate the data heterogeneity which is crucial for our later theorems.

Assumption 1. *(Lipschitz continuity). The loss function ℓ satisfies the following Lipschitz smoothness conditions: $\|\ell(\theta_1, z) - \ell(\theta_2, z)\| \leq L\|\theta_1 - \theta_2\|$, $\|\nabla\ell(\theta_1, z) - \nabla\ell(\theta_2, z)\| \leq L_\theta\|\theta_1 - \theta_2\|$, and $\|\nabla\ell(\theta, z_1) - \nabla\ell(\theta, z_2)\| \leq L_z\|z_1 - z_2\|_p$, where ∇ is the abbreviation for ∇_θ used throughout the paper.*

Note that Assumption 1 is widely used in existing studies (Xing, Song, and Cheng 2021a; Li, Song, and Yang 2023; Zhu et al. 2023; Kanai et al. 2023). Intuitively, different local distributions affect the global population risk and hence may affect the model generalization as well. To effectively measure the exacerbated heterogeneity of client i in AL, we account for both their original data distribution P_i and the distribution of adversarially generated samples $\tilde{P}_i$.

Lemma 1. *Under Assumption 1 and given $i \in [m]$, for any θ we have*

$$\|\nabla R_i(\theta) - \nabla R(\theta)\| \leq (2\rho L_z + 6L)D_i,$$

where $D_i = \max\{d_{TV}(\tilde{P}_i, P_i), d_{TV}(P_i, P), d_{TV}(\tilde{P}, P)\}$.

Remark 1. The total variation distance d_{TV} is used to compare these distributions against their respective global counterparts, P and $\tilde{P}$. We define D_i as the maximum total variation observed among these comparisons. This metric captures the extent of data variation from both regular and adversarial perspectives, with higher values indicating greater heterogeneity. This lemma reveals that when AL tries to gain more robustness through stronger adversarial generation, the heterogeneity will be exacerbated. When $\rho = 0$, we have $d_{TV}(\tilde{P}_i, P_i) = 0, d_{TV}(\tilde{P}, P) = 0$, thus D_i is the same as (Sun, Niu, and Wei 2023) in FL. In particular, we define $D_{\max} = \max_{i\in[m]} D_i$ to denote the maximum heterogeneity among all clients in the FAL framework.

Vanilla FAL Algorithm

To approach the global objective in (3), (Zizzo et al. 2020; Shah et al. 2021; Li et al. 2022; Zhang et al. 2022) designed the Vanilla FAL (VFAL) algorithm. The main idea of VFAL is to leverage AL in FedAvg (McMahan et al. 2017) locally. Each client runs on a local copy of the global model θ^t with its local data to conduct AL. Then, the server receives updated model parameters $\{\theta_{i,K}^{t+1}\}_{i=1}^m$ for all clients and performs the following aggregation: $\theta^{t+1} = \frac{1}{m}\sum_{i=1}^{m}\theta_{i,K}^{t+1}$, where K is the epoch of local training. The parameters θ^{t+1} for the global model are then sent back to each client for another epoch of training. Next, we use the on-average bound to derive the generalization error in the VFAL algorithm through the following theorem.

Theorem 1. *If a VFAL algorithm $\mathcal{A}$ is ϵ-on-averagely stable, we can obtain the generalization error $\varepsilon_{gen}(\mathcal{A})$ as follows:*

$$\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[\frac{1}{m}\sum_{i=1}^{m}(R_i(\mathcal{A}(\mathcal{S})) - R_{\mathcal{S}_i}(\mathcal{A}(\mathcal{S})))\right] \leq \epsilon.$$

Most existing studies analyze the generalization error of algorithm stability under smooth loss functions (Sun, Niu, and Wei 2023; Huang et al. 2023; Lei, Sun, and Liu 2023), exploring the dependence of FL generalization properties on heterogeneity. However, (Liu et al. 2020) suggest that the adversarial loss $\ell_\rho(\theta, z)$ remains non-smooth, even if we assume the standard loss $\ell(\theta, z)$ is smooth. This non-smoothness violates some basic properties in stability analysis (Hardt, Recht, and Singer 2016; Lei, Sun, and Liu 2023), bringing additional challenges.

To address this issue, a natural approach is to use smoothness approximation techniques, such as (1) *Surrogate Smoothness Approximation (SSA)* (Cui et al. 2021), (2) *Randomized Smoothness Approximation (RSA)* (Alashqar et al. 2023), and (3) *Over-Parameterized Smoothness Approximation (OPSA)* (Li, Song, and Yang 2023; Li et al. 2022). Beyond data heterogeneity in FAL, non-smoothness in approximation can affect the generalization error, further complicating the problem within the FAL framework. Therefore, we aim to investigate the relationship between generalization error and these three smoothness approximation methods. This will help in clearly understanding the problem and in designing more efficient algorithms for FAL.

Surrogate Smoothness Approximation

Using the surrogate smoothness helps improve the quality of the gradient of the original function ℓ_ρ, potentially improving generalization (Xie et al. 2020). In particular, we reconsider the following surrogate loss to substitute ℓ_ρ in (3):

$$h(\theta; z_{i,j}) = \max_{\|z_{i,j} - z'_{i,j}\|_p \leq \rho} \ell(\theta; z'_{i,j}), \tag{5}$$

and each client performs $\theta^t_{i,k+1} = \theta^t_{i,k} - \eta_t \nabla h(\theta^t_{i,k}, z_{i,j})$, where $k \in [K]$ and η_t is the local stepsize at the global epoch t. Based on the surrogate loss h, we have a set of properties of SSA, which can be defined as follows:

Definition 3. *Let $\beta \geq 0, \xi \geq 0$ and $h(\theta)$ be a differentiable function. We say $h(\theta)$ is ξ-approximately β-gradient Lipschitz, if $\forall\ \theta_1$ and θ_2, we have*

$$\|\nabla h(\theta_1) - \nabla h(\theta_2)\| \leq \beta \|\theta_1 - \theta_2\| + \xi, \ \ \xi = 2\rho z.$$

From Definition 3, we can see that the SSA method dynamically inherits the non-smooth properties by AL. If $\rho = 0$, h is gradient Lipschitz; otherwise, $\rho > 0$, h is a general non-smooth function. In particular, the maximization operation of the surrogate function h improves the continuity of the gradient and helps smooth out potentially sharp gradients.

Theorem 2. *Let the step size be chosen as $\eta_t \leq \frac{1}{\beta K(t+1)}$. Under Assumption 1, the generalization bound ε_{gen} with the SSA method satisfies:*

$$\mathcal{O}\left(\rho T \log T + \frac{T\sqrt{\log T \Delta}}{m n_{\min}} + \frac{T \log T (\rho+1) D_{\max}}{m n_{\min}}\right),$$

where $\Delta = \mathbb{E}[R(\theta_0)] - \mathbb{E}[R(\theta^*)]$ and $n_{\min} = \min_{i\in[m]} n_i$.

Remark 2. As shown in Theorem 2, the first term, $\rho T \log T$, represents the approximation error from SSA, which is affected by ρ of AL. We can see that a smaller ρ helps reduce the error in this dominant term. The second term $\frac{T\sqrt{\log T \Delta}}{m n_{\min}}$ pertains to the algorithm's convergence performance. However, in practice, the third term, $\frac{T \log T (\rho+1) D_{\max}}{m n_{\min}}$, arises from the heterogeneity, which quantifies the variation between the original and adversarial data. This term reveals the increased heterogeneity leads to poorer robust generalization ability. When $\rho = 0$, the first term disappears, while the third still reflects the original heterogeneity, similar to FL (Sun, Niu, and Wei 2023).

Randomized Smoothness Approximation

In RSA, we consider a smoothed alternative version of ℓ_ρ in (3) based on the randomized smoothing technique (Duchi, Bartlett, and Wainwright 2012; Lin, Zheng, and Jordan 2022). Specifically, let $\gamma \geq 0, \ell_\gamma(\theta, z_{i,j}) = \mathbb{E}_{u\sim\mathbb{P}^d}[\ell_\rho(\theta + \gamma u_{i,j}, z_{i,j})]$, where γ is a smoothing parameter, $\mathbb{P}$ is an uniform distribution on a unit ball in d-dimensional space with the $\mathcal{L}_2$-norm, and $u_{i,j} \sim \mathbb{P}^d$ is a random variable generated from $\mathbb{P}^d$. By adding multiple instances of randomized noise to the parameters, we aim to steer the new objective function, ℓ_γ, towards a flatter weight loss landscape compared to the original objective ℓ_ρ, which can moderate the non-smooth property (Neyshabur et al. 2017; Kanai et al. 2023).

In practice, the gradient of $\ell_\gamma(\theta)$ is difficult to compute due to the expectation. Therefore, we can estimate it using Markov chain Monte-Carlo techniques (Vrugt et al. 2009) on each client i as follows:

$$\theta^t_{i,k+1} = \theta^t_{i,k} - \frac{\eta_t}{Q} \sum_{q=1}^{Q} \nabla \ell_\rho(\theta^t_{i,k} + \gamma u_{i,k_q}, z_{i,k}). \tag{6}$$

Note that in each local epoch $k \in [K]$, u_{i,k_q} is sampled from $\mathbb{P}^d$ locally, and we need to repeat the number of Q times samples to reduce the estimated error between real gradient and expected gradient (Duchi, Bartlett, and Wainwright 2012; Lin, Zheng, and Jordan 2022).

Theorem 3. *Let $c \geq 0$ be a constant and the step size be chosen as $\eta_t \leq \frac{\gamma}{4K\sqrt{d}cL(t+1)}$. Under Assumption 1, the generalization bound ε_{gen} with the RSA method satisfies:*

$$\mathcal{O}\left(\frac{T^{1/4}\log T}{\sqrt{Q}} + \frac{T^{3/4}\sqrt{\Delta}}{m n_{\min}} + \frac{T((\rho+1)D_{\max})^{1/3}}{m n_{\min}}\right),$$

where $\Delta = \mathbb{E}[R(\theta_0)] - \mathbb{E}[R(\theta^*)]$ and $n_{\min} = \min_{i\in[m]} n_i$.

Remark 3. Similar to Theorem 2, the generalization error bound of RSA is also composed of three terms. We can see the first term, i.e., $\frac{T^{\frac{1}{4}}\log T}{\sqrt{Q}}$, where $\frac{1}{\sqrt{Q}}$ represents the discrepancy in the gradient estimate between real gradient and expected gradient, is independent on ρ. Note that this term is smaller as Q increases, but a larger Q imposes an unbearable computational cost on clients. Hence, the adjustment of Q should be appropriately in a small range. In addition, since each client optimizes a new smoothed objective function, ℓ_γ, the second term $\frac{T^{\frac{3}{4}}\sqrt{\Delta}}{m n_{\min}}$ and the third term $\frac{T((\rho+1)D_{\max})^{\frac{1}{3}}}{m n_{\min}}$ are generated based on a flatter weight space compared to the original. This is why they are independent of Q and the order of T is smaller.

Over-parameterized Smoothness Approximation

In OPSA (Lei, Jin, and Ying 2022), we focus on the following shallow neural network of the form in (1):

$$f_W(x) := \sum_{\tau=1}^{s} \mu_\tau \varphi(\langle w_\tau, x\rangle), \tag{7}$$

where we fix $\mu_\tau \in \{-1/\sqrt{s}, 1/\sqrt{s}\}, \varphi : \mathbb{R} \mapsto \mathbb{R}$ is an activation function and $W = (w_1, \ldots, w_s) \in \mathbb{R}^{d\times s}$ is the weight

matrix. In (7), w_τ denotes the weight of the edge connecting the input to the τ-th hidden node, and μ_τ is the weight of the edge connecting the τ-th hidden node to the output node. Here s is the number of nodes in the hidden layer, commonly referred as the width of an over-parameterized neural network. Although the network's width is crucial, it does not conflict with the model parameters θ as a vector, since θ represents the entire set of model parameters, including the weight matrix W and other parameters. The difference lies only in the form of parameter representation. Thus, we have $f_\theta = f_W$ and $\ell_\rho(\theta, z) = \ell_\rho(W, z)$, demonstrating that both notations fundamentally describe the same model parameterization. Following the studies (Richards and Kuzborskij 2021; Lei, Jin, and Ying 2022; Wang et al. 2023), we state two standard assumptions as follows:

Assumption 2. *(Activation). The activation $\varphi(\cdot)$ is continuous and twice differentiable with constant $B_\varphi, B_{\varphi'}, B_{\varphi''} \geq 0$ bounding $|\varphi(\cdot)| \leq B_\varphi, |\varphi'(\cdot)| \leq B_{\varphi'}$ and $|\varphi''(\cdot)| \leq B_{\varphi''}$.*

Assumption 3. *(Inputs, labels, parameters, and the loss function). For constants $C_x, C_y, C_W, C_0 > 0$, inputs belong to $\mathcal{B}_2^d(C_x)$, labels belong to $[-C_y, C_y]$, the weight matrix W are confined within the bounded domain $\mathcal{B}_2^d(C_W)$, and the loss is uniformly bounded by C_0 almost surely.*

In over-parameterized network, we can bound the Hessian scales of ℓ_ρ, i.e., $\|\nabla^2 \ell_\rho\| \leq \zeta_\theta$, by using the above assumptions to ensure its ζ_θ-smoothness, where $\zeta_\theta = 2(C_x^2 + \rho^2)\big(B_{\varphi'}^2 + B_{\varphi''} B_\varphi + \frac{B_{\varphi''} C_y}{\sqrt{s}}\big)$ (Lei, Jin, and Ying 2022; Wang et al. 2023; Sitawarin, Chakraborty, and Wagner 2020).

Theorem 4. *Without loss of generality, we assume $4K\eta_t C_0 \geq 1$ and $s \geq 16\eta_t^2 T^2 K^2 (b' H_K)^2 (1 + 2\eta_t \zeta_\theta)^2$,where $b' = C_x^2 B_{\varphi''}(C_x B_{\varphi'} + \sqrt{2C_0}), H_K = 2\sqrt{K\eta_t C_0}$. Let the step size be chosen as $\eta_t \leq \frac{1}{6K\zeta_\theta}$. Under Assumptions 1-3, the generalization bound ε_{gen} with the OPSA method satisfies:*

$$\mathcal{O}\left(\frac{T^{\frac{1}{2}}\sqrt{\Delta}}{m n_{\min}} + \frac{T(\rho^2\sqrt{s}+1) D_{\max}}{m n_{\min}}\right),$$

where $\Delta = \mathbb{E}[R(\theta_0)] - \mathbb{E}[R(\theta^*)]$ and $n_{\min} = \min_{i\in[m]} n_i$.

Remark 4. In Theorem 4, the approximation error appearing in Theorems 2-3 is eliminated due to the smoothness of ℓ_ρ under over-parameterization. In addition, the first term, $\frac{T^{\frac{1}{2}}\sqrt{\Delta}}{m n_{\min}}$, is smaller because over-parameterization helps the model avoid the complex non-convex area during the optimization process, allowing algorithms like gradient descent to find the optimal solutions (Arora, Cohen, and Hazan 2018) more quickly. However, hidden behind this benefit is the curse of width exacerbation, denoted as $\mathcal{O}(\rho^2\sqrt{s})$. This is similar to the findings in (Wu et al. 2021; Hassani and Javanmard 2024), where the curse of width $\mathcal{O}(\rho\sqrt{s})$ was observed to impact perturbation stability in AL. Therefore, it is necessary to control for the exacerbation of heterogeneity by the width of the network.

Remark 5. Based on the results in Theorems 2-4, RSA is the most effective method for reducing the generalization error by constructing a higher-quality mediator function. In additional, by mitigating the attack strength of ρ, we can improve the dominate term $\mathcal{O}(\rho T \log T)$ in SSA. The results of OPSA consist of two terms, but it has the largest generalization error due to the presence of $\sqrt{s}$, which leads to the highest order term of $\mathcal{O}(T^2)$. Note that all three theorems include an additional term related to the number of K local training epochs. However, due to the lower order, e.g., $\frac{T\sqrt{\log T}}{m n_{\min} K^{1/2}}$ of SSA in Theorem 2, we dismiss these terms as they do not have significant impacts.

Slack FAL Algorithm

From the above results, we can see how to set smoothness approximation to reduce the generalization error in degrading VFAL algorithm (Zizzo et al. 2020; Shah et al. 2021; Hong et al. 2021). Instead of the VFAL algorithm, (Zhu et al. 2023) proposed a simple but effective SFAL algorithm based on an α-slack decomposed mechanism.

α-Slack Decomposed Mechanism

In the VFAL algorithm, the simple averaging aggregation strategy often causes the global model to be biased towards clients with significant data heterogeneity. In contrast, the SFAL algorithm effectively corrects this bias by dynamically reweighting the contributions of clients through the α-slack decomposed mechanism. In particular, the α-slack decomposed mechanism uses the local adversarial loss $R_{\mathcal{S}_i}(\theta)$ as an AL-related metric to identify more heterogeneous clients and weaken their weights, thereby emphasizing the importance of those less heterogeneous clients.

Given $\alpha \in [0, 1), \hat{m} \leq \frac{m}{2}$ with the local empirical risk sorted by $\{R_{\mathcal{S}_i}(\theta)\}$ in ascending order, we have $\sum_{i=1}^{\hat{m}} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta) \leq \sum_{i=\hat{m}+1}^{m} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)$, where $\phi_{(\cdot)}$ is a function which maps the index to original empirical risk. Then we modified the original global objective (3) as follows:

$$R_{\mathcal{S}}(\theta) = \frac{1}{\tilde{m}}\Bigg[(1+\alpha)\sum_{i=1}^{\hat{m}} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta) + (1-\alpha)\sum_{i=\hat{m}+1}^{m} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)\Bigg], \tag{8}$$

where $\tilde{m} = (1+\alpha)\hat{m} + (1-\alpha)(m-\hat{m})$. Note that $\tilde{m}$ is introduced to ensure that the weights of each local risk are normalized in the global aggregation. To flexibly emphasize the importance of partial populations, we have

$$\alpha = 1 - \frac{\sum_{i=1}^{\hat{m}} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)}{\sum_{i=\hat{m}+1}^{m} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)}. \tag{9}$$

In particular, if partial populations $\sum_{i=1}^{\hat{m}} R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)$ is smaller, we need a larger α to enhance their contribution in aggregation. At the high level, we assign the client-wise slack during aggregation to upweight the clients having the small AL loss (simultaneously downweight those with large loss). Moreover, we would like to emphasize that this α-slack mechanism does not affect the result of Theorem 1, which is the key theorem in our proof of the generalization bound using on-average stability.

App. Method	VFAL	SFAL
SSA	$\mathcal{O}(\rho T\log T+\frac{T\sqrt{\log T\Delta}}{mn_{\min}}+\frac{T\log T(\rho+1)D_{\max}}{mn_{\min}})$	$\mathcal{O}(\rho T\log T+\frac{T\sqrt{\log T\Delta}}{r_\alpha mn_{\min}}+\frac{T\log T(\rho+1)D_{\max}}{r_\alpha mn_{\min}})$
RSA	$\mathcal{O}(\frac{T^{\frac{1}{4}}\log T}{\sqrt{Q}}+\frac{T^{\frac{3}{4}}\sqrt{\Delta}}{mn_{\min}}+\frac{T((\rho+1)D_{\max})^{\frac{1}{3}}}{mn_{\min}})$	$\mathcal{O}(\frac{T^{\frac{1}{4}}\log T}{\sqrt{Q}}+\frac{T^{\frac{3}{4}}\sqrt{\Delta}}{r_\alpha mn_{\min}}+\frac{T((\rho+1)D_{\max})^{\frac{1}{3}}}{r_\alpha mn_{\min}})$
OPSA	$\mathcal{O}(\frac{T^{\frac{1}{2}}\sqrt{\Delta}}{mn_{\min}}+\frac{T(\rho^2\sqrt{s}+1)D_{\max}}{mn_{\min}})$	$\mathcal{O}(\frac{T^{\frac{1}{2}}\sqrt{\Delta}}{r_\alpha mn_{\min}}+\frac{T(\rho^2\sqrt{s}+1)D_{\max}}{r_\alpha mn_{\min}})$

Table 1: Main theoretical results.

Theorem 5. *If a SFAL algorithm $\mathcal{A}$ is ϵ-on-averagely stable, we can obtain the generalization error $\varepsilon_{gen}(\mathcal{A})$ as follows:*

$$\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[\frac{1+\alpha}{\tilde{m}}\sum_{i=1}^{\hat{m}}(R_i^{\phi_{(i)}}(\mathcal{A}(\mathcal{S}))-R_{\mathcal{S}_i}^{\phi_{(i)}}(\mathcal{A}(\mathcal{S})))\right]+$$
$$\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[\frac{1-\alpha}{\tilde{m}}\sum_{i=\hat{m}+1}^{m}(R_i^{\phi_{(i)}}(\mathcal{A}(\mathcal{S}))-R_{\mathcal{S}_i}^{\phi_{(i)}}(\mathcal{A}(\mathcal{S})))\right]\leq\epsilon.$$

Remark 6. Theorem 5 implying that under SFAL, analyzing generalization using the algorithmic stability is more challenging. Given the dynamic reweighting nature of the α-slack mechanism, we need to consider two scenarios in which the client i, holding the perturbed sample $z'_{i',j}$ where defined in Definition 1, experiences dynamic reweighting across different global aggregation epochs:(1) upweighting, when $i\in[1,\hat{m}]$, and (2) downweighting, when $i\in[\hat{m}+1,m]$.

Generalization Bound of SFAL

Here, we present the generalization bound of the three smoothness approximation methods in SFAL.

Theorem 6. *Given $\alpha\in[0,1),\hat{m}\leq\frac{m}{2},r_\alpha=1+\frac{\alpha}{1-\alpha}\frac{2\hat{m}}{m}$, we can obtain the following results in SFAL:*
1. Under Assumption 1, let the step size be chosen as $\eta_t\leq\frac{\tilde{m}}{m\beta K(t+1)}$, the generalization bound ε_{gen} with the SSA method satisfies:

$$\mathcal{O}\left(\rho T\log T+\frac{T\sqrt{\log T\Delta}}{r_\alpha mn_{\min}}+\frac{T\log T(\rho+1)D_{\max}}{r_\alpha mn_{\min}}\right).$$

2. Under Assumption 1, let $c\geq0$ be a constant and the step size be chosen as $\eta_t\leq\frac{\tilde{m}\gamma}{4mK\sqrt{d}cL(t+1)}$. Under Assumption 1, the generalization bound ε_{gen} with the RSA method satisfies:

$$\mathcal{O}\left(\frac{T^{\frac{1}{4}}\log T}{\sqrt{Q}}+\frac{T^{\frac{3}{4}}\sqrt{\Delta}}{r_\alpha mn_{\min}}+\frac{T((\rho+1)D_{\max})^{\frac{1}{3}}}{r_\alpha mn_{\min}}\right).$$

3. Without loss of generality, we assume $4K\eta_tC_0\geq1$ and $s\geq16\eta_t^2T^2K^2(b'H_K)^2(1+2\eta_t\zeta_\theta)^2$,where $b'=C_x^2B_{\varphi''}(C_xB_{\varphi'}+\sqrt{2C_0}),H_K=2\sqrt{K\eta_tC_0}$. Let the step size be chosen as $\eta_t\leq\frac{\tilde{m}}{6mK\zeta_\theta}$. Under Assumptions 1-3, the generalization bound ε_{gen} with the OPSA method satisfies:

$$\mathcal{O}\left(\frac{T^{\frac{1}{2}}\sqrt{\Delta}}{r_\alpha mn_{\min}}+\frac{T(\rho^2\sqrt{s}+1)D_{\max}}{r_\alpha mn_{\min}}\right),$$

where $\Delta=\mathbb{E}[R(\theta_0)]-\mathbb{E}[R(\theta^*)]$ and $n_{\min}=\min_{i\in[m]}n_i$.

Remark 7. In Theorem 6, as $\alpha\to1$, r_α also becomes larger due to the corresponding increase in $\frac{\alpha}{1-\alpha}$, effectively mitigating the generalization error. When $\alpha=0$, it yields the result of the VFAL algorithm. Moreover, (9) implies that there is a balance between α and $\hat{m}$. When $\hat{m}$ increase, $\sum_{i=1}^{\hat{m}}R_{\mathcal{S}_i}^{\phi_{(i)}}(\theta)$ also increases, which means we need to decrease α to avoid overemphasizing the importance of partial populations. Combining the results of VFAL and SFAL, we obtain Table 1. Note that the α-slack mechanism operates on the server side, while the three smoothness approximation methods operate on the client side. We observe that both the term including Δ and the term including $D_{\max}$ in SFAL, which are generated from the aggregation process, are optimized compared to VFAL. Therefore, SFAL enhances the generalization of these methods without altering their strengths and weaknesses.

Remark 8. Combining the analysis of the above results, we gain some insights that help to design more efficient FAL algorithms. Motivated by SFAL, we propose designing a new contrastive loss for weight assignment to measure heterogeneity exacerbation. Based on Lemma 1, the local data drift $D_i=d_{TV}(\tilde{P}_i,P_i)$ should be the dominant term in $D_{\max}$. Therefore, we consider a new dynamic aggregation strategy, i.e., $R_{\mathcal{S}}(\theta)=\sum_{i=1}^{m}\frac{d_{TV}(\tilde{P}_i,P_i)}{\sum_{i=1}^{m}d_{TV}(\tilde{P}_i,P_i)}R_{\mathcal{S}_i}(\theta)$. Concretely, at the client level, we can approximate $d_{TV}(\tilde{P}_i,P_i)$ in terms of the generated adversarial samples and original samples. We believe this aggregation strategy effectively combines FL and AL. For local training, we may design a new adversarial penalty, i.e., $\|\theta_{\text{adv}}-\theta_{\text{ori}}\|$, to address heterogeneity $D_{\max}$ caused by AL. Here, θ_{adv} is trained on adversarial samples, and θ_{ori} is trained on original samples. Compared to SFAL, our proposed algorithms integrate information from the original samples, which may not only improve the generalization ability for adversarial samples but also have the potential to enhance the ability for non-adversarial samples.

Experiments

Setups. Here, we conduct the experiments on the SVHN dataset (Netzer et al. 2011) under heterogeneous scenarios. Following (Shah et al. 2021), we partition the dataset based on labels using a skew parameter a. This distribution allows each of the m clients to receive samples from certain classes. We define $\mathcal{Y}$ as the set of all classes, equally divided among m clients to form $\mathcal{Y}_\varrho$. Each client thus holds $(100-(m-1)\times a)\%$ of its data from classes in $\mathcal{Y}_\varrho$ and $a\%$ from other classes. The generalization gap is measured as the difference between train and test accuracy. For the inner problems, we

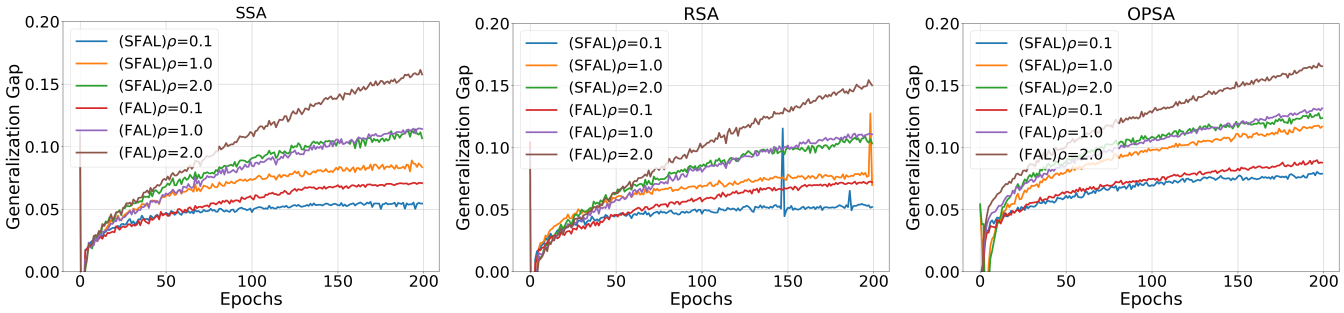

Figure 1: Generalization Gap of the attack strength ρ on SVHN. ($m = 40, a = 2.0$)

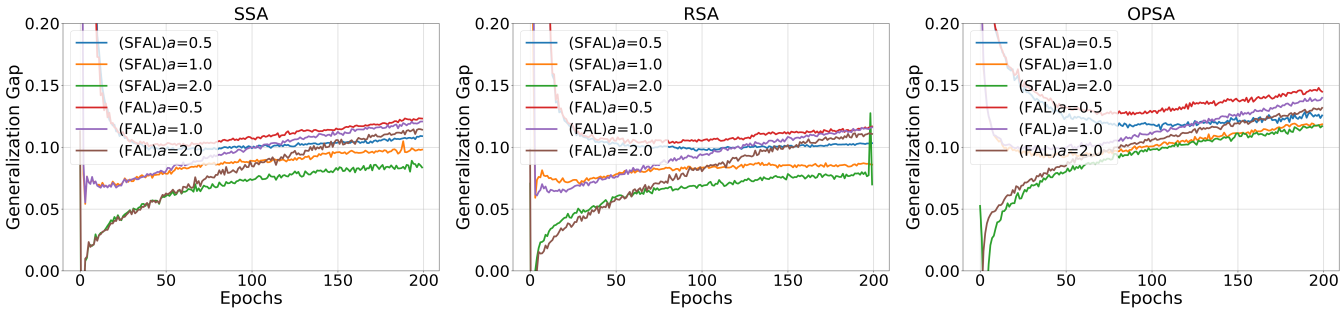

Figure 2: Generalization Gap of the skew parameter a on SVHN. ($m = 40, \rho = 1.0$)

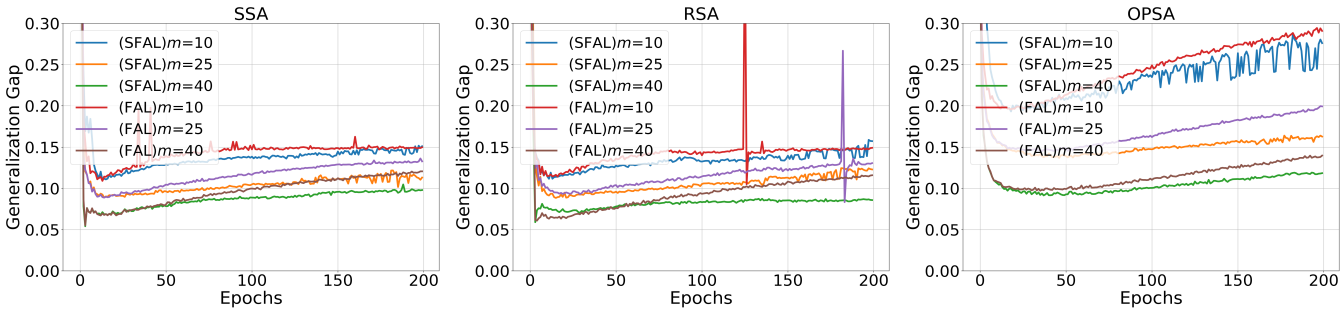

Figure 3: Generalization Gap of the number of client m on SVHN. ($a = 1.0, \rho = 1.0$)

adopt the ℓ_∞ PGD adversarial training in (Madry et al. 2018), the step size in the inner maximization is set to be $\rho/4$. In SFAL, we set $\hat{m} = m/5$. More detailed settings and other results are provided in supplementary.

Impact on ρ. Comparing different ρ in Figure 1, we can see that the generalization gap increases as ρ gets larger, which indicate that stronger attacks produce greater heterogeneity in our bound, i.e., $\rho D_{\max}$.

Impact on a. Comparing different a in Figure 2, we can find that the enhanced heterogeneity of FL itself similarly increases the generalization gap, which is consistent with Lemma 1, i.e., $D_i = d_{TV}(P_i, P)$.

Impact on m. Comparing different m in Figure 3, we can infer that increasing the number of clients can effectively reduce the generalization gap since the effect of heterogeneity on generalization can be dispersed.

In summary, RSA achieves the minimum generalization error compared to OPSA and SSA. Additionally, under suitable α, SFAL effectively reduces the generalization error compared to VFAL.

Conclusion

In this paper, using three different smoothness approximation methods: SSA, RSA, and OPSA, we provide generalization upper bounds for two popular FAL algorithms: VFAL and SFAL. Our bounds explicitly explain how to set proper parameters in these approximation methods and which one is more helpful in reducing the generalization error for FAL. In general, the RSA method can achieve the best generalization performance. We also find that SFAL always performs better than SFAL due to its re-weighted aggregation strategy. Based on our analysis, we give some useful insights into designing more efficient and dynamic global aggregation strategies to mitigate heterogeneity under the FAL context.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2021YFF1201300), the Project of Xiangjiang Laboratory (24XJJCYJ01003), the National Natural Science Foundation of China (62202293, 62372472, 62341201, 62320106006, 62302525), Beijing Natural Science Foundation (L231012), the Key Research and Development Program of Hunan Province of China (2023SK2020), the Hunan Provincial Natural Science Foundation of China (2024JJ4068, 2024JJ6527), and the High Performance Computing Center of Central South University.

References

Alashqar, B.; Gasnikov, A.; Dvinskikh, D.; and Lobanov, A. 2023. Gradient-free federated learning methods with l 1 and l 2-randomization for non-smooth convex stochastic optimization problems. *Computational Mathematics and Mathematical Physics*, 63(9): 1600–1653.

Arora, S.; Cohen, N.; and Hazan, E. 2018. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International conference on machine learning*, 244–253. PMLR.

Athalye, A.; Carlini, N.; and Wagner, D. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, 274–283. PMLR.

Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; and Shmatikov, V. 2020. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, 2938–2948. PMLR.

Bousquet, O.; and Elisseeff, A. 2002. Stability and generalization. *The Journal of Machine Learning Research*, 2: 499–526.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. Ieee.

Cheng, X.; Fu, K.; and Farnia, F. 2024. Stability and Generalization in Free Adversarial Training. *arXiv preprint arXiv:2404.08980*.

Croce, F.; and Hein, M. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, 2206–2216. PMLR.

Cui, S.; Pan, W.; Liang, J.; Zhang, C.; and Wang, F. 2021. Addressing algorithmic disparity and performance inconsistency in federated learning. *NeurIPS*, 34: 26091–26102.

Duchi, J. C.; Bartlett, P. L.; and Wainwright, M. J. 2012. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2): 674–701.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Hardt, M.; Recht, B.; and Singer, Y. 2016. Train faster, generalize better: Stability of stochastic gradient descent. In *International conference on machine learning*, 1225–1234. PMLR.

Hassani, H.; and Javanmard, A. 2024. The curse of overparametrization in adversarial training: Precise analysis of robust generalization for random features regression. *The Annals of Statistics*, 52(2): 441–465.

Hong, J.; Wang, H.; Wang, Z.; and Zhou, J. 2021. Federated robustness propagation: Sharing adversarial robustness in federated learning. *arXiv preprint arXiv:2106.10196*, 1.

Hu, X.; Li, S.; and Liu, Y. 2022. Generalization bounds for federated learning: Fast rates, unparticipating clients and unbounded losses. In *The Eleventh International Conference on Learning Representations*.

Huang, W.; Shi, Y.; Cai, Z.; and Suzuki, T. 2023. Understanding convergence and generalization in federated learning through feature learning theory. In *The Twelfth International Conference on Learning Representations*.

Kanai, S.; Yamada, M.; Takahashi, H.; Yamanaka, Y.; and Ida, Y. 2023. Relationship between nonsmoothness in adversarial training, constraints of attacks, and flatness in the input space. *IEEE Trans. Neural Netw. Learn. Syst.*

Kearns, M.; and Ron, D. 1997. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. In *Proceedings of the tenth annual conference on Computational learning theory*, 152–162.

Kuzborskij, I.; and Lampert, C. 2018. Data-dependent stability of stochastic gradient descent. In *International Conference on Machine Learning*, 2815–2824. PMLR.

Lei, Y.; Jin, R.; and Ying, Y. 2022. Stability and generalization analysis of gradient methods for shallow neural networks. *NeurIPS*, 35: 38557–38570.

Lei, Y.; Sun, T.; and Liu, M. 2023. Stability and Generalization for Minibatch SGD and Local SGD. *arXiv preprint arXiv:2310.01139*.

Li, X.; Qu, Z.; Tang, B.; and Lu, Z. 2023. Fedlga: Toward system-heterogeneity of federated learning via local gradient approximation. *IEEE Trans. Cybern.*, 54(1): 401–414.

Li, X.; Qu, Z.; Zhao, S.; Tang, B.; Lu, Z.; and Liu, Y. 2021. Lomar: A local defense against poisoning attack on federated learning. *IEEE Trans. Dependable Secure Comput*, 20(1): 437–450.

Li, X.; Song, Z.; Tao, R.; and Zhang, G. 2022. A convergence theory for federated average: Beyond smoothness. In *International Conference on Big Data*, 1292–1297. IEEE.

Li, X.; Song, Z.; and Yang, J. 2023. Federated adversarial learning: A framework with convergence analysis. In *International Conference on Machine Learning*, 19932–19959. PMLR.

Lin, T.; Zheng, Z.; and Jordan, M. 2022. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *NeurIPS*, 35: 26160–26175.

Liu, C.; Salzmann, M.; Lin, T.; Tomioka, R.; and Süsstrunk, S. 2020. On the loss landscape of adversarial training: Identifying challenges and how to overcome them. *NeurIPS*, 33: 21476–21487.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.

Mohri, M.; Sivek, G.; and Suresh, A. T. 2019. Agnostic federated learning. In *International Conference on Machine Learning*, 4615–4625. PMLR.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A. Y.; et al. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, 4. Granada.

Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; and Srebro, N. 2017. Exploring generalization in deep learning. *NeurIPS*, 30.

Qu, Z.; Li, X.; Duan, R.; Liu, Y.; Tang, B.; and Lu, Z. 2022. Generalized federated learning via sharpness aware minimization. In *International conference on machine learning*, 18250–18280. PMLR.

Qu, Z.; Li, X.; Han, X.; Duan, R.; Shen, C.; and Chen, L. 2023. How to Prevent the Poor Performance Clients for Personalized Federated Learning? In *Proceedings of the IEEE/CVF CVPR*, 12167–12176.

Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečnỳ, J.; Kumar, S.; and McMahan, H. B. 2020. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*.

Rice, L.; Wong, E.; and Kolter, Z. 2020. Overfitting in adversarially robust deep learning. In *International conference on machine learning*, 8093–8104. PMLR.

Richards, D.; and Kuzborskij, I. 2021. Stability & generalisation of gradient descent for shallow neural networks without the neural tangent kernel. *NeurIPS*, 34: 8609–8621.

Sadeghi, A.; Wang, G.; Ma, M.; and Giannakis, G. B. 2020. Learning while respecting privacy and robustness to distributional uncertainties and adversarial data. *arXiv preprint arXiv:2007.03724*.

Samangouei, P.; Kabkab, M.; and Chellappa, R. 2018. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models. In *International Conference on Learning Representations*.

Shah, D.; Dube, P.; Chakraborty, S.; and Verma, A. 2021. Adversarial training in communication constrained federated learning. *arXiv preprint arXiv:2103.01319*.

Shalev-Shwartz, S.; Shamir, O.; Srebro, N.; and Sridharan, K. 2010. Learnability, stability and uniform convergence. *The Journal of Machine Learning Research*, 11: 2635–2670.

Sitawarin, C.; Chakraborty, S.; and Wagner, D. 2020. SAT: Improving Adversarial Training via Curriculum-Based Loss Smoothing. *arXiv preprint arXiv:2003.09347*.

Sun, Z.; Niu, X.; and Wei, E. 2023. Understanding generalization of federated learning via stability: Heterogeneity matters. *arXiv preprint arXiv:2306.03824*.

Sun, Z.; and Wei, E. 2022. A communication-efficient algorithm with linear convergence for federated minimax learning. *NeurIPS*, 35: 6060–6073.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Vrugt, J. A.; ter Braak, C. J.; Diks, C. G.; Robinson, B. A.; Hyman, J. M.; and Higdon, D. 2009. Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International journal of nonlinear sciences and numerical simulation*, 10(3): 273–290.

Wang, H.; Sreenivasan, K.; Rajput, S.; Vishwakarma, H.; Agarwal, S.; Sohn, J.-y.; Lee, K.; and Papailiopoulos, D. 2020. Attack of the tails: Yes, you really can backdoor federated learning. *NeurIPS*, 33: 16070–16084.

Wang, P.; Lei, Y.; Wang, D.; Ying, Y.; and Zhou, D.-X. 2023. Generalization Guarantees of Gradient Descent for Multi-Layer Neural Networks. *arXiv preprint arXiv:2305.16891*.

Wu, B.; Chen, J.; Cai, D.; He, X.; and Gu, Q. 2021. Do wider neural networks really help adversarial robustness? *NeurIPS*, 34: 7054–7067.

Xiao, J.; Fan, Y.; Sun, R.; Wang, J.; and Luo, Z.-Q. 2022a. Stability analysis and generalization bounds of adversarial training. *NeurIPS*, 35: 15446–15459.

Xiao, J.; Zhang, J.; Luo, Z.-Q.; and Ozdaglar, A. E. 2022b. Smoothed-SGDmax: A Stability-Inspired Algorithm to Improve Adversarial Generalization. In *NeurIPS ML Safety Workshop*.

Xie, C.; Tan, M.; Gong, B.; Yuille, A.; and Le, Q. V. 2020. Smooth adversarial training. *arXiv preprint arXiv:2006.14536*.

Xing, Y.; Song, Q.; and Cheng, G. 2021a. On the algorithmic stability of adversarial training. *NeurIPS*, 34: 26523–26535.

Xing, Y.; Song, Q.; and Cheng, G. 2021b. On the generalization properties of adversarial training. In *International Conference on Artificial Intelligence and Statistics*, 505–513. PMLR.

Zhang, G.; Lu, S.; Zhang, Y.; Chen, X.; Chen, P.-Y.; Fan, Q.; Martie, L.; Horesh, L.; Hong, M.; and Liu, S. 2022. Distributed adversarial training to robustify deep neural networks at scale. In *Uncertainty in Artificial Intelligence*, 2353–2363. PMLR.

Zhang, J.; Li, B.; Chen, C.; Lyu, L.; Wu, S.; Ding, S.; and Wu, C. 2023. Delving into the adversarial robustness of federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 11245–11253.

Zhou, Y.; Wu, J.; Wang, H.; and He, J. 2022. Adversarial robustness through bias variance decomposition: A new perspective for federated learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2753–2762.

Zhu, J.; Yao, J.; Liu, T.; Yao, Q.; Xu, J.; and Han, B. 2023. Combating exacerbated heterogeneity for robust models in federated learning. *arXiv preprint arXiv:2303.00250*.

Zizzo, G.; Rawat, A.; Sinn, M.; and Buesser, B. 2020. Fat: Federated adversarial training. *arXiv preprint arXiv:2012.01791*.